Correspondences of the Third Kind:
Camera Pose Estimation from Object Reflection

Kohei Yamashita[1], Vincent Lepetit[2], and Ko Nishino[1]

[1] Graduate School of Informatics, Kyoto University, Kyoto, Japan
kyamashita@vision.ist.i.kyoto-u.ac.jp kon@i.kyoto-u.ac.jp
https://vision.ist.i.kyoto-u.ac.jp/
[2] LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-vallée, France
vincent.lepetit@enpc.fr

Abstract. Computer vision has long relied on two kinds of correspondences: pixel correspondences in images and 3D correspondences on object surfaces. Is there another kind, and if there is, what can they do for us? In this paper, we introduce correspondences of the third kind we call reflection correspondences and show that they can help estimate camera pose by just looking at objects without relying on the background. Reflection correspondences are point correspondences in the reflected world, *i.e.*, the scene reflected by the object surface. The object geometry and reflectance alter the scene geometrically and radiometrically, respectively, causing incorrect pixel correspondences. Geometry recovered from each image is also hampered by distortions, namely generalized bas-relief ambiguity, leading to erroneous 3D correspondences. We show that reflection correspondences can resolve the ambiguities arising from these distortions. We introduce a neural correspondence estimator and a RANSAC algorithm that fully leverages all three kinds of correspondences for robust and accurate joint camera pose and object shape estimation just from the object appearance. The method expands the horizon of numerous downstream tasks, including camera pose estimation for appearance modeling (*e.g.*, NeRF) and motion estimation of reflective objects (*e.g.*, cars on the road), to name a few, as it relieves the requirement of overlapping background.

Keywords: Camera Pose · Reflection · Bas-Relief Ambiguity

1 Introduction

Look at the two images in Fig. 1. Even for this extreme case of a perfect mirror object, we—as humans—can understand how the camera moved (at least qualitatively) between the two images. This likely owes to our ability to disentangle the reflected surroundings from surface appearance.

How would a computer estimate the camera pose change between the two images? Structure-from-motion would fail in such conditions because they rely on pixel correspondences, *i.e.*, matched projections of the same physical surface points in the images, which would be erroneous as the glossy reflection violates the color constancy constraint.

Recent neural shape-from-shading methods can recover the object geometry as surface normals in each view even under complex natural illumination [30]. As the illumination is unknown, the recovered surface normals, however, suffer from a fundamental ambiguity known as the generalized bas-relief ambiguity [1] between light source and surface geometry. That is, a rotated illumination and sheared surface can conspire to generate the exact same object appearance. As such, 3D correspondences established between the two surface normal maps recovered independently from the two images would only tell us the camera pose in the distorted space.

Fig. 1: We humans can tell how the camera moved between the images, but computers have a hard time. Can we estimate camera pose and possibly object shape just from object appearance, despite the featureless appearance and non-overlapping background? (Photos by Richard Ellis/Alamy)

In fact, we show that even if we use both pixel correspondences in the images and 3D correspondences on the recovered object surfaces, we cannot resolve this ambiguity for orthographic cameras. Even with a perspective camera, the object is often far enough that this limitation still holds. Even if we have more than two views, joint estimation of the camera poses and the object shape with the erroneous normal maps would be still challenging. The geometry estimate can easily fall into local minima whose surface normals (*i.e.*, local surface geometry) are consistent with the current camera pose estimates.

How then can the camera pose be estimated just from the object appearance? We take a hint from what we humans likely do: establish correspondences in the world reflected by the object surface. By leveraging the single-view reconstruction of surface normals, we can extract from each of the images a reflectance map. The reflectance map is the Gaussian sphere of the reflected radiance. It represents the surrounding environment modulated by surface reflectance as a spherical surface indexed by surface normals. We establish correspondences between the reflectance maps of the images. We refer to these as reflection correspondences.

We show that these reflection correspondences by themselves are not sufficient to directly compute the camera pose. They, however, resolve the ambiguity remaining in the pixel and 3D correspondences due to the bas-relief ambiguity. This means that, by using all three types of correspondences in two or more images of an object of arbitrary reflectance, even without any texture and with strong specularity, we can compute the relative camera poses just from the object appearance. This liberates many applications from the seemingly benign yet practically extremely limiting requirement of overlapping static background or diffuse surface texture just to recover camera positions.

We formalize reflection correspondences and show how they should be combined with conventional correspondences (pixel correspondences and 3D corre-

spondences) to obtain a quantitative estimate of the correct camera rotation. We introduce a RANSAC-based, two-step algorithm which first exploits conventional correspondences and then resolves the ambiguity with reflection correspondences. We also introduce a new neural feature extractor for establishing 3D and reflection correspondences robust to the inherent distortions primarily caused by the bas-relief ambiguity with effective data augmentation. Finally, we derive a joint estimation framework for accurate joint camera pose and geometry reconstruction which alternates between the interdependent two quantities.

Experimental results on synthetic and real images show that the reflection correspondences and the neural feature extractor as well as our iterative estimation framework are essential for accurate camera pose and geometry estimation. We believe reflection correspondences can play an important role in applications beyond camera pose and shape recovery from object appearance, including camera calibration [2] and object pose estimation [22] when classical correspondences are not sufficient. All data and code can be found on our project web page.

2   Related Work

In this paper, we address camera pose estimation from a small number (2) of views of textureless, non-Lambertian (*e.g.*, shiny) objects taken under unknown natural illumination, which remains challenging for existing methods especially when good initial estimates are not available.

Structure-from-motion methods recover camera poses for multiple images by detecting and leveraging pixel correspondences [25, 35]. They typically detect such pixel correspondences by leveraging view-independent, salient features such as textures and jointly solve for camera poses and 3D locations of the surface points. Descriptors invariant to rotation and scale (*e.g.*, SIFT [16]) and outlier correspondence detection with RANSAC [6] are often used for robust estimation. However, these methods are prone to fail on textureless, non-Lambertian objects (especially, shiny objects) as correspondences between surface points are extremely challenging to establish on these objects. Also, correspondences based on textures are usually sparse, and these methods require a large number (*e.g.*, 50) of input images that have large visual overlaps for accurate estimation.

Neural image synthesis methods jointly estimate surface geometry and a surface light field (or its radiometric roots, *i.e.*, reflectance and illumination) as neural representations with differentiable volumetric [15, 18, 20, 27, 34] or surface [31, 33] rendering. While most of these methods require multiple registered images as inputs, a few methods also handle non-registered multi-view images by optimizing the camera poses as additional parameters [3, 31]. They, however, still require good initial estimates of camera poses (*e.g.*, SAMURAI [3] uses manually annotated coarse camera poses as inputs) or an extremely large number (>50) of input images as the joint optimization of all unknown parameters easily falls into local minima.

Pose estimation from specular reflection has also been studied [5, 8, 12, 14, 19, 24]. Lagger *et al.* [12], for instance, refine camera pose estimates using im-

(a) Pixel Correspondences   (b) 3D Correspondences   (c) Reflection Correspondences
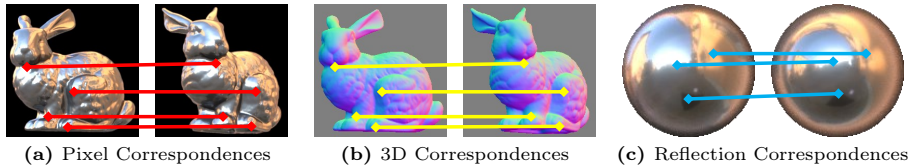
Fig. 2: We show how to leverage three types of correspondences. *Pixel correspondences* (a) are the pixels that correspond to the same surface point. We can also leverage similar correspondences in the normal maps which we refer to as *3D correspondences* (b). In addition to these correspondences, we leverage novel correspondences about the surrounding environment which we can observe through surface reflection. We recover camera-view reflectance maps, maps that associate surface normal orientations with the surrounding environment, and detect this type of correspondences from them (c). We refer to this novel kind of correspondences as **reflection correspondences**.

ages and the 3D CAD model of a specular object. They recover view-dependent environment maps from the inputs and optimize the camera pose by minimizing discrepancies between them. These methods, however, assume known geometry [5, 12, 19, 24], known illumination [5, 8, 14], or a simple lighting model [19, 24] which cannot be assumed in general scenes.

In contrast, reflection correspondences enable us to recover the camera pose and object geometry of arbitrary objects under complex natural illumination without requiring static overlapping background to be present in the image.

3   Method

In this section, we first detail the three types of correspondences we consider and the equations that can be derived from them. Figure 2 provides a visualization of these three types of correspondences. To avoid confusion, we give distinct names to them.

- "pixel correspondences" for the classical correspondences which link corresponding locations in the two images on the object surface: the two matched points are the reprojections of the same physical 3D point in the two images.
- "3D correspondences" for the correspondences between corresponding normals in the two images on the object surface. Note that for each pixel correspondence, we also have a 3D correspondence if we know the normals at the matched image locations. The difference is that for the pixel correspondence, we exploit the pixel coordinates themselves, while in the case of the 3D correspondence, we exploit the normals.
- "reflection correspondences" which match image locations where *light rays mirror-reflected by the object surface come from the same direction in the two images*. As shown in Fig. 2(c), we detect these correspondences from camera-view reflectance maps, *i.e.*, view-dependent maps about the surrounding environment.
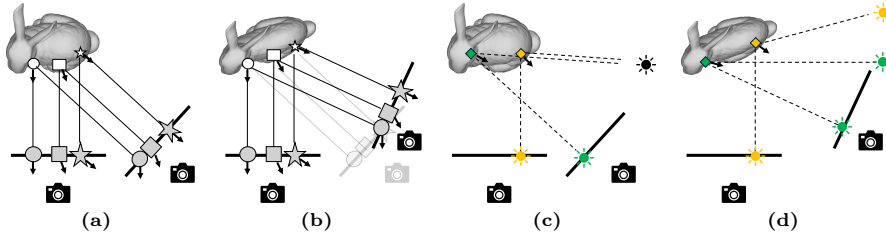
Fig. 3: As depicted in (a) and (b), due to the generalized bas-relief ambiguity in single-view surface normal recovery [1] and the fundamental difficulty in structure-from-motion [9], we cannot obtain a unique solution for the relative rotation from pixel and 3D correspondences when the cameras are orthographic (*e.g.*, distant from the object). Reflection correspondences, *i.e.*, correspondences regarding the incident directions for specular reflections, enable us to distinguish the correct relative rotation (c) from the other possible solutions like (d).

In this paper, we assume orthographic projection as objects are usually distant from the cameras and we can regard camera rays that point at the object as almost constant. Note that, as we see in the next subsection, camera pose estimation is challenging especially under orthographic projection.

3.1 Pixel Correspondences

Let us assume that we can establish $N_\mathcal{I}$ pixel correspondences $(u_1^i, v_1^i) \leftrightarrow (u_2^i, v_2^i)$. Under orthographic projection, each pixel correspondence gives us one constraint on the camera rotation between the two images [9]:

$$\left[(u_1^i \cos\phi + v_1^i \sin\phi) - (u_2^i \cos\theta + v_2^i \sin\theta) \right]^2 = 0, \tag{1}$$

where ϕ and θ are two of the three Euler angles (θ, ϕ, η) that correspond to the relative rotation R_{21} from the second view to the first one. Without loss of generality, we assume that these correspondences are translated such that the average location (*e.g.*, $\frac{1}{N_\mathcal{I}} \sum_i u_1^i$) is zero. We use the (z-x-z) sequence for the Euler angles

$$R_{21} = R_z[\phi] \, R_x[\eta] \, R_z[-\theta], \tag{2}$$

where $R_z[\cdot]$ and $R_x[\cdot]$ are rotations around z and x axes, respectively. As depicted in Figs. 3a and 3b, since Eq. (1) is independent of one of the unknown Euler angles (η), the rotation matrix cannot be fully recovered from pixel correspondences of only two views.

3.2 3D Correspondences

If we could estimate unambiguously the normal maps, $\mathbf{N}_1$ and $\mathbf{N}_2$, for the images, correspondences $(u_1^i, v_1^i) \leftrightarrow (u_2^i, v_2^i)$ would be constrained by

$$\mathbf{N}_1[u_1^i, v_1^i] = R_{21}\mathbf{N}_2[u_2^i, v_2^i], \tag{3}$$

as the normals at locations (u_1^i, v_1^i) and (u_2^i, v_2^i) are the same up to rotation R_{21}. We then could estimate the relative rotation R_{21} by simply solving Eq. (3).

Unfortunately, in practice, normal maps can be recovered only up to the generalized bas-relief (GBR) ambiguity [1]. Given a normal map $\mathbf{N}$ for an image I, any normal map $\mathbf{N}'$ with

$$\mathbf{N}'[u, v] = G^{-T}\mathbf{N}[u, v], \tag{4}$$

could result in the same image I under a different lighting, where G is the generalized bas-relief (GBR) transformation [1]

$$G \equiv \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \mu & \nu & \lambda \end{pmatrix}. \tag{5}$$

The parameters μ and ν can take any values, and λ can take any positive value. Note that, although discussions regarding the bas-relief ambiguity (*e.g.*, one in Belhumeur *et al.* [1]) usually assume directional lights, this holds true even for environmental illumination as we can view it as a set of directional lights.

The bas-relief ambiguity changes the relationship in Eq. (3) to

$$\mathbf{N}_1[u_1^i, v_1^i] \propto G_1^{-T}R_{21}G_2^{T}\mathbf{N}_2[u_2^i, v_2^i], \tag{6}$$

where G_k ($k = 1, 2$) is an unknown GBR transformation for each view that corresponds to the estimation errors [1]. If we have a sufficient number of correspondences, we can obtain a unique solution for the combined transformation

$$G_{21} \equiv G_1^{-T}R_{21}G_2^{T}. \tag{7}$$

There is, however, an unresolvable ambiguity in its decomposition into the three unknown matrices. Most important, this ambiguity corresponds to one regarding the pixel correspondences. In other words, we cannot recover the relative camera pose even when combining pixel and 3D correspondences: for any η, there are corresponding GBR transformations G_1 and G_2 that are consistent with Eq. (6). We provide the proof in the supplementary material.

3.3   Reflection Correspondences

Let us assume we have correspondences between image locations $(u_1^i, v_1^i) \leftrightarrow (u_2^i, v_2^i)$ where *light rays come from the same direction in the two images*. In other words, correspondences of the surroundings reflected by the object surface. If we, for now, ignore the GBR ambiguity, these "reflection correspondences" give each a constraint of the form

$$\omega_r(\mathbf{N}_1[u_1^i, v_1^i]) \propto R_{21}\omega_r(\mathbf{N}_2[u_2^i, v_2^i]), \tag{8}$$

where function $\omega_r(\mathbf{n})$ returns the reflection of the line of sight direction ω_o on the surface with normal $\mathbf{N}$:

$$\omega_r(\mathbf{n}) = -\omega_o + 2\frac{\omega_o \cdot \mathbf{n}}{\mathbf{n} \cdot \mathbf{n}}\mathbf{n}. \tag{9}$$

Again, at this stage, we can in fact predict the normal maps only up to the GBR ambiguity, and we need to introduce the two GBR transformations into Eq. (8):

$$\omega_{\mathrm{r}}(\mathrm{G_1}^{\mathrm{T}}\mathbf{N}_1[u_1^i, v_1^i]) \propto \mathrm{R}_{21}\omega_{\mathrm{r}}(\mathrm{G_2}^{\mathrm{T}}\mathbf{N}_2[u_2^i, v_2^i]). \tag{10}$$

As illustrated in Figs. 3c and 3d, each reflection correspondence thus gives us a new type of equation to estimate R_{21} but also G_1 and G_2, from which we can also get the object shape.

Detecting these reflection correspondences directly from images is, however, extremely challenging as surface reflection depends not only on the surrounding illumination but also on the surface geometry. Let us now assume we have a surface normal map for each view. Then we can avoid this problem by recovering camera-view reflectance maps [10]. The reflectance maps are view-dependent mappings from a surface normal to the surface radiance which are determined by the surface reflectance and the surrounding illumination environment

$$E_k(\mathbf{n}) = \int L_i(\mathrm{R}_k^{\mathrm{T}}\omega_{\mathbf{i}})\psi(\omega_{\mathbf{i}}, \omega_{\mathbf{o}}, \mathbf{n})\max(\omega_{\mathbf{i}} \cdot \mathbf{n}, 0)\mathrm{d}\omega_{\mathbf{i}}, \tag{11}$$

where $\omega_{\mathbf{i}}$, $\omega_{\mathbf{o}}$, and $\mathbf{n}$ are incident, viewing, and surface normal orientations in the local camera coordinate system, respectively. R_k is the camera pose (a rotation matrix) of k-th view, $L_i(\mathrm{v})$ is a mapping from an incident direction to a radiance of the corresponding incident light, and $\psi(\omega_{\mathbf{i}}, \omega_{\mathbf{o}}, \mathbf{n})$ is the bidirectional reflectance distribution function (BRDF).

From the surface normal maps and the input images, $i.e.$, pairs of surface normals and surface radiances, we can recover these reflectance maps [23, 30] and, as illustrated in Fig. 2(c), detect these reflection correspondences from the reflectance maps regardless of the object shape. Note that, as the reflectance maps are recovered using the surface normal maps, they also suffer from the bas-relief ambiguity. The relationship between the ground truth $E(\mathbf{n})$ and another possible solution $E'(\mathbf{n})$ is

$$E'(\mathbf{n}) = E(\mathrm{G}^{\mathrm{T}}\mathbf{n}). \tag{12}$$

Thus we still need to use Eq. (10) for these reflection correspondences.

3.4   Relative Rotation from Correspondences

From Eq. (1), (6), and (10), we can derive an objective function

$$f = f_{\mathcal{I}} + f_{\mathcal{N}}^{(12)} + f_{\mathcal{N}}^{(21)} + f_{\mathcal{R}}^{(12)} + f_{\mathcal{R}}^{(21)}. \tag{13}$$

that enforces the equation in the least-squares sense. $f_{\mathcal{I}}$ enforces Eq. (1):

$$f_{\mathcal{I}} = \frac{1}{N_{\mathcal{I}}}\sum_i^{N_{\mathcal{I}}} \left(t_\phi^i - t_\theta^i\right)^2, \tag{14}$$

with

$$t_\phi^i = u_1^i \cos\phi + v_1^i \sin\phi, \tag{15}$$

$$t_\theta^i = u_2^i \cos\theta + v_2^{(i)} \sin\theta\,, \tag{16}$$

and $N_\mathcal{I}$ is the number of image correspondences.

Note that for Eq. (6) and Eq. (10), we can switch the role of the two images. We therefore introduce two terms for each of these equations. For Eq. (6), we introduce

$$f_\mathcal{N}^{(jk)} = \frac{1}{N_\mathcal{N}} \sum_i^{N_\mathcal{N}} \left\| \mathbf{N}_j[u_j^i, v_j^i] - \mathrm{Norm}(\mathrm{G}_{kj} \mathbf{N}_k[u_k^i, v_k^i]) \right\|^2 , \tag{17}$$

where Norm is the vector normalization operator and $N_\mathcal{N}$ the number of 3D correspondences.

For Eq. (10), we introduce

$$f_\mathcal{R}^{(jk)} = \frac{1}{N_\mathcal{R}} \sum_i^{N_\mathcal{R}} \left\| \mathbf{N}_j[u_j^i, v_j^i] - \Omega^{(jk)}(\mathbf{N}_k[u_k^i, v_k^i]) \right\|^2 , \tag{18}$$

where $\Omega^{(jk)}(\mathbf{n})$ transforms a surface normal $\mathbf{n}$ in the j-th view reflectance map to the surface normal in the k-th view

$$\Omega^{(jk)}(\mathbf{n}) = \mathrm{Norm}(\mathrm{G}_k^{-T} \omega_\mathrm{r}^{-1}(\mathrm{R}_{jk}\, \omega_\mathrm{r}(\mathrm{G}_j^T \mathbf{n})))\,. \tag{19}$$

Given 3 sets of $N_\mathcal{I}$ pixel correspondences, $N_\mathcal{N}$ 3D correspondences, and $N_\mathcal{R}$ reflection correspondences, we can optimize f in Eq. (13) for the three Euler angles θ, ϕ, and η, and the two sets of parameters for the GBR transformations for both images $\mu_1, \nu_1, \lambda_1, \mu_2, \nu_2, \lambda_2$, under the constraints that λ_1 and λ_2 are positive.

Naive optimization with all the correspondences can easily be affected by erroneous correspondences which are unavoidable. We also empirically find that it is slow and susceptible to local minima especially due to the nonlinear operation $\omega_\mathrm{r}(\cdot)$. We instead derive a RANSAC-based, two-step algorithm that first estimates the combined transformation G_{21} using pixel and 3D correspondences and then decomposes it using reflection correspondences.

In the first step, given the pixel and the 3D correspondences, our algorithm recovers the combined transformation $\mathrm{G}_{21} \equiv \mathrm{G}_1^{-T} \mathrm{R}_{21} \mathrm{G}_2^T$ based on RANSAC [6]. For each of the L_1 iterations, we build a set consisting of randomly sampled M_1 pixels and M_1 3D correspondences. In practice, we set M_1 to 4. We empirically find this to be the smallest necessary number of pixel and 3D correspondences. Please see the supplementary material for details. We hope to derive a theoretical justification of this number in our future work. For each set, we obtain an estimate of G_{21} by minimizing an objective function f_1 using an off-the-shelf solver for nonlinear optimization [4,26]. f_1 is similar to the sum of $f_\mathcal{I}$, $f_\mathcal{N}^{(12)}$, and $f_\mathcal{N}^{(21)}$ in Eq. (13), though we compute f using only the sampled correspondences. We then select a good estimate $\hat{\mathrm{G}}_{21}$ from them according to the number of pixels and 3D correspondences that are consistent with the estimate.
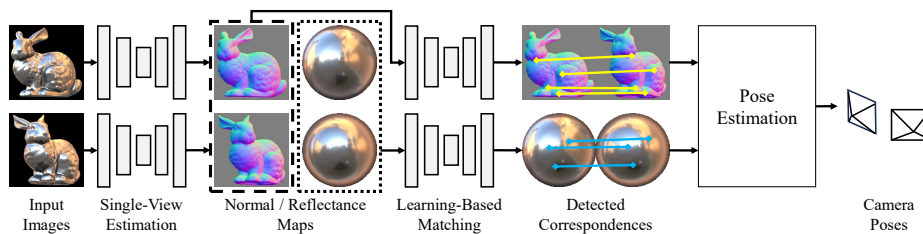
Input | Single-View | Normal / Reflectance | Learning-Based | Detected | | Camera
Images | Estimation | Maps | Matching | Correspondences | | Poses

Fig. 4: Given two-view images of a textureless, non-Lambertian object, we first recover the surface normals and a reflectance map for each view using a single-view geometry reconstruction method [30]. We establish 3D and reflection correspondences with a novel deep feature extraction network to compute the relative camera pose from them.

In the second step, given the estimate $\hat{G}_{21}$ and the reflection correspondences, we obtain a combination of G_1, G_2, and R_{21} that is consistent with $\hat{G}_{21}$ and most of the reflection correspondences. The key idea here is that, if we determine η, one of the three Euler angles of R_{21} that remains ambiguous, we can decompose $\hat{G}_{21}$ into G_1, G_2, and R_{21} uniquely and analytically. Based on this, for each possible η, we compute the corresponding G_1, G_2, and R_{21}, and select a solution that maximizes the number of reflection correspondences that are consistent with the decomposed transformations. The supplementary material provides more details and pseudo code.

Translation Estimation Once R_{21} is recovered, by leveraging inlier pixel correspondences, we can also solve for the relative translation between the two views except for offsets regarding the two viewing directions [9]. That is, we estimate the translation vector $\mathbf{t_{21}} = (t_x, t_y, t_z)$ using the constraint

$$r_{23}t_x - R_{13}t_y = -\sum_i \left(r_{23}\tilde{u}^i - r_{13}\tilde{v}^i \right) ,\tag{20}$$

where r_{jk} is (j, k) element of R_{21} and

$$\tilde{u}^i = r_{11}u_2^i + r_{12}v_2^i - u_1^i ,\tag{21}$$

$$\tilde{v}^i = r_{21}u_2^i + r_{22}v_2^i - v_1^i .\tag{22}$$

3.5   Camera Pose Estimation from Two Images

Based on the algorithm in Sec. 3.4, we can derive a method, as depicted in Figure 4, that takes in only two-view images of a textureless, non-Lambertian object and estimates the relative camera pose (the relative rotation and the relative translation) between the two views. We achieve this by recovering a surface normal and a reflectance map for each view, detecting the correspondences

from the recovered maps, and applying the proposed algorithm to the detected correspondences.

To recover the normal and reflectance maps for each view separately, we use DeepShaRM [30] which is a radiometry-based geometry estimation method. DeepShaRM can jointly recover geometry, reflectance maps, and normal maps from an image by alternating between (1) learning-based reflectance map estimation from images and a geometry estimate, (2) learning-based surface normal estimation from images and the estimated reflectance maps, and (3) geometry optimization with the estimated surface normals. This gives us the normal and reflectance maps but only up to the GBR transformations. Please see the supplementary material for more implementation details.

To obtain 3D correspondences in the normal maps, we train a convolutional neural network to extract pixel-wise, view-invariant features that we use to match surface points between normal maps. We also train another deep neural network for the detection of reflection correspondences in the same way. We train the network by contrastive learning. We use pairs of synthetic normal maps of two views as training data. We feed both of them to the feature extraction network and obtain corresponding feature maps $\mathbf{F}_1$ and $\mathbf{F}_2$. Given ground truth correspondences $(u_1^i, v_1^i) \leftrightarrow (u_2^i, v_2^i)$, we impose the InfoNCE loss [21]

$$L = \sum_i \sum_j -\delta_{ij} \log \left(\frac{\exp(c_{ij})}{\sum_{i'} \sum_{j'} \exp(c_{i'j'})} \right), \qquad (23)$$

where c_{ij} is the cosine similarity of feature vectors $\mathbf{F}_1[u_1^i, v_1^i]$ and $\mathbf{F}_2[u_2^j, v_2^j]$

$$c_{ij} = \mathrm{Norm}(\mathbf{F}_1[u_1^i, v_1^i]) \cdot \mathrm{Norm}(\mathbf{F}_2[u_2^j, v_2^j]), \qquad (24)$$

and δ_{ij} is the Kronecker delta. By this, we ensure that features of pixels that correspond to the same surface point become similar.

The training with synthetic normal maps is, however, insufficient in practice as the surface normal maps estimated by the single-view estimation are distorted by GBR transform. We overcome this with data augmentation. For each view in the training data, we randomly sample parameters of the GBR transformation and transform the input normal map according to Eq. (4). We use the transformed normal maps instead of the original ones so that the network can learn to extract features robust to the estimation errors caused by the bas-relief ambiguity.

At inference time, using the extracted feature maps, we detect correspondences by brute-force matching and filter them with the ratio test by Lowe [16]. For any location on the object in the first image (or reflectance map), we look for the best and second best locations in the second image (or reflectance map) in terms of the cosine similarity of the features. If the ratio of the similarity of the second best location to one of the best location is lower than a threshold, we use the best location as the matched location. This gives us all of pixel, 3D, and reflection correspondences.

Table 1: (a) Mean camera pose (relative rotation) estimation accuracy on images in the nLMVS-Synth dataset [29]. (b) Camera pose estimation accuracy on real-world images. (*) COLMAP [25] uses 11 view uncropped images as inputs.

	Pose Error
w/o Data Augm.	7.4 deg
w/o Joint	6.2 deg
w/o RM	6.6 deg
Ours	4.5 deg

(a) Synthetic Result

	Planck	Horse	Bunny	Cat
COLMAP [25] (*)	15.7 deg	N/A	24.5 deg	N/A
LightGlue [13]	5.0 deg	18.9 deg	93.6 deg	35.4 deg
SAMURAI [3]	5.6 deg	24.3 deg	30.5 deg	68.7 deg
Ours	1.9 deg	12.5 deg	7.2 deg	8.6 deg

(b) Real Data Result

Once we have the three types of correspondences, we can recover the relative rotation and the relative translation using the two-step estimation algorithm (Sec. 3.4).

3.6   Joint Shape and Camera Pose Recovery

Once we obtain the relative camera pose, we can further improve its accuracy by consolidating the multi-view surface normal estimates using the camera pose estimates and exploiting the improved surface normals and corresponding reflectance maps to update the camera pose estimate. As DeepShaRM [30] is originally designed for posed multi-view images, we achieve this by alternating between multi-view surface normal and reflectance map estimation by DeepShaRM and camera pose estimation by our method. As a byproduct of DeepShaRM, we can also obtain an accurate object shape.

4   Experimental Results

We focus our experiments on answering the following key questions. How well do the reflection correspondences contribute to the camera pose estimation accuracy? Can the novel feature extraction network establish correspondences robust to GBR distortion? How does our method generalize to real-world images? How does our method compare to structure-from-motion methods and neural image synthesis methods with camera pose optimization? To answer these questions, we validate the following points

- The proposed components (especially the reflection correspondences) are essential for accurate reconstruction from a sparse set (2) of images.
- Our method can recover accurate camera poses even from two real images.

Training Data We trained the deep networks in our method (including DeepShaRM [30]) on the training set of the nLMVS-Synth dataset [29]. The training set consists of 26850 images of 2685 synthetic objects. The training shapes are composed of primitive shapes (ellipsoids, cubes, cylinders) augmented with random height fields [28]. 94 materials and 2685 environmental maps from existing databases [7, 17, 32] are used for rendering.
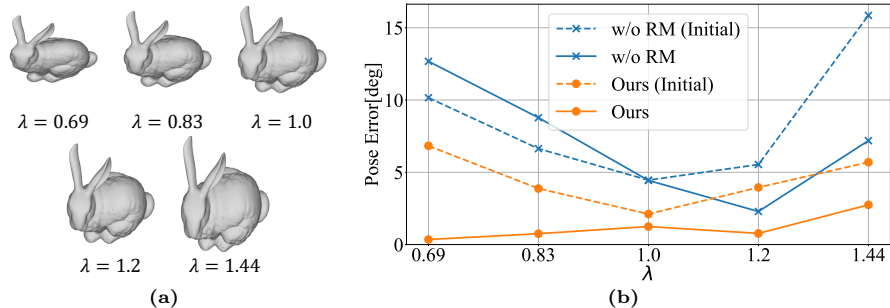
Fig. 5: Evaluation on synthetic shapes with different levels of flatness. (a) We create such shapes by applying GBR transformations with different λ. (a) Accuracy for each λ. The results shows the effectiveness of using reflection correspondences. Please see the text for details.

4.1 Evaluation on Synthetic Data

We evaluate the accuracy of the joint camera pose and object shape estimation framework on synthetic images rendered with unseen shapes, BRDFs, and illumination maps. We used synthetic images from the test set of the nLMVS-Synth dataset [29] and some rendered by ourselves for this evaluation. In total, the test set consists of 27 combinations of 3 shapes, 3 BRDFs, and 3 illumination environments.

We evaluate the accuracy of our method by computing the geodesic distance between the estimated relative rotation and the ground truth. Note that evaluation of the estimated translation under orthographic projection is difficult due to unsolvable offsets regarding the two viewing directions. We compare our camera pose estimation accuracy with those by its own ablated variants, "w/o Data Augm.", "w/o Joint," and "w/o RM.". "w/o Data Augm." uses the feature extraction networks trained without the data augmentation with GBR transforms. "w/o Joint" is our method without the joint iterative estimation which recovers camera poses from only the initial estimates of normal and reflectance maps. "w/o RM" ignores the bas-relief ambiguity and recovers the relative rotation using only pixel and 3D correspondences based on Eqs. (1) and (3).

Table 1a shows the average camera pose estimation errors. The results show that the joint estimation, the data augmentation method, and the use of reflection correspondences are essential for accurate camera pose estimation.

4.2 Robustness to Various Levels of Flatness

To further clarify the effectiveness of the reflection correspondences, we also test our method on synthetic shapes with different levels of flatness. As shown in Figure 5a, we created such test shapes by applying GBR transformations with different parameters to the Stanford Bunny [11]. We set μ and ν in the GBR transformations to be zero and λ to be one of 0.69, 0.83, 1.0, 1.2, and 1.44. Using
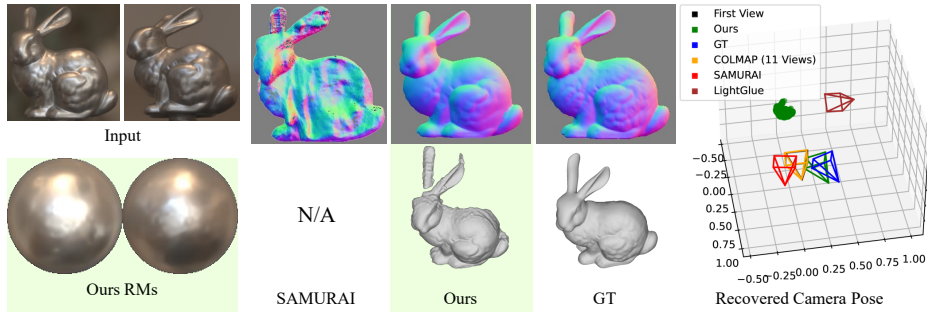
Fig. 6: Reflectance maps (RMs), normal maps, surface geometry, and relative camera poses recovered from two-view real-world images [29].
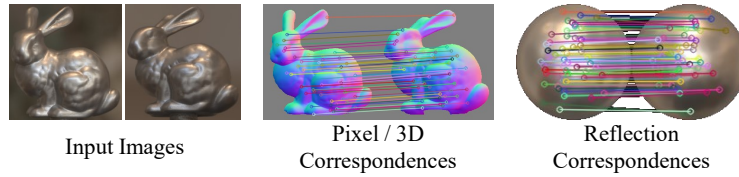

Fig. 7: All types of correspondences detected from the real image pair for Fig. 6.

the distorted shapes along with BRDF data and illumination maps used in the synthetic evaluation above, we rendered synthetic images for this evaluation. We tested our method and the baseline method that does not exploit reflection correspondences ("w/o RM") on them.

Figure 5b shows the mean camera pose estimation error for each λ, *i.e.*, each level of flatness. The baseline method that exploits only conventional correspondences works well only for a shape with a "normal" level of flatness (*i.e.*, close to $\lambda = 1$). This is because the method heavily relies on a realistic prior learned by the deep geometry estimation method [30]. In contrast, our method is robust to various levels of flatness which is critical for practical use in the real-world.

4.3   Evaluation on Real Data

Quantitative evaluation of our method on in-the-wild images is difficult as they are by definition difficult to register with traditional methods. Nevertheless, we were able to evaluate our method on the real images from the nLMVS-Real dataset [29] and some captured and calibrated by ourselves.

We applied our method trained on the synthetic data to images of four real objects without any fine tuning. We compared our results with those of applying COLMAP [25], LightGlue [13], and SAMURAI [3]. We found that COLMAP completely fails on our inputs, *i.e.*, two-view cropped images that cover only the target object. For images from the nLMVS-Real, we used 11 view uncropped images that capture not only the target object but also textured object around the

target (*i.e.*, ChArUco boards) as inputs to COLMAP. Note that the uncropped images are provided by the authors of the dataset [29] as RAW data.

Table 1b and Fig. 6 show quantitative and qualitative results. Note that COLMAP completely failed on the "Horse" object. As SAMURAI failed to extract a 3D mesh model from their volumetric geometry representation, we only show a normal map for SAMURAI. In contrast to these existing methods which fail on these challenging inputs, our method successfully recovers plausible camera poses, surface geometry, and reflectance maps.

Scenes with textured and textureless objects We can also leverage our method to estimate camera poses from sparse images capturing scenes consisting of both textured and textureless, non-Lambertian objects often found in the real world. In such a setting, the pixel correspondences can come from the textured object and the reflection correspondences from the textureless object. That is, the different types of correspondences can come from different objects in the scene to recover the unique camera poses capturing all the objects in each shot.

Figure 8 shows qualitative result of our method and that of a baseline structure-from-motion (SfM) method on an example scene. Note that, although the cameras are almost orthographic, we can still use a perspective model to obtain a unique estimate of the relative rotation from only the pixel correspondences. The results show that, while the SfM method (orange) struggles to recover the angle between the two views (*i.e.*, η), our camera pose estimation result (green) is almost identical to the ground truth (blue). This demonstrates the effectiveness of our method even in scenes consisting of multiple different types of objects. Please see the supplementary material for more results.
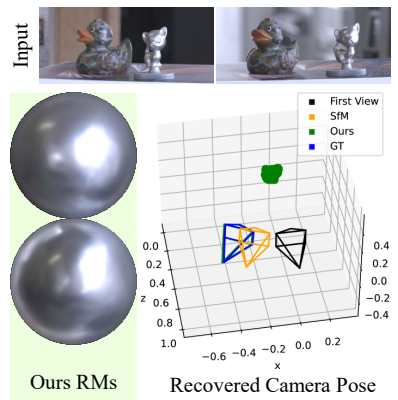

Fig. 8: Results on images capturing scenes consisting of both textured and textureless, non-Lambertian objects. Our result aligns almost perfectly with the ground truth.

5 Conclusion

We introduced a new type of correspondences, reflection correspondences, which enables accurate and robust camera pose and joint shape estimation from the appearance alone of textureless, reflective objects. We believe this new type of correspondences not only expunges restricting requirements on image capture for camera pose estimation but also opens new use of object appearance in various applications. We hope to explore these in future work and also catalyze research for this with our release of code and data.

References

1. Belhumeur, P.N., Kriegman, D.J., Yuille, A.L.: The Bas-Relief Ambiguity. IJCV **35**(1), 33–44 (1999)
2. Bhayani, S., Sattler, T., Larsson, V., Heikkilä, J., Kukelova, Z.: Partially Calibrated Semi-Generalized Pose from Hybrid Point Correspondences. In: Proc. WACV (2023)
3. Boss, M., Engelhardt, A., Kar, A., Li, Y., Sun, D., Barron, J.T., Lensch, H.P.A., Jampani, V.: SAMURAI: Shape And Material from Unconstrained Real-world Arbitrary Image collections. In: Proc. NeurIPS (2022)
4. Branch, M.A., Coleman, T.F., Li, Y.: A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. SIAM J. Sci. Comput. **21**(1), 1–23 (1999)
5. Chang, J.Y., Raskar, R., Agrawal, A.K.: 3D Pose Estimation and Segmentation using Specular Cues. In: Proc. CVPR. pp. 1706–1713. IEEE Computer Society (2009)
6. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Commun. ACM **24**(6), 381–395 (1981)
7. Gardner, M.A., Sunkavalli, K., Yumer, E., Shen, X., Gambaretto, E., Gagné, C., Lalonde, J.F.: Learning to Predict Indoor Illumination from a Single Image. ACM TOG **36**(6), 176:1–176:14 (2017)
8. Han, K., Liu, M., Schnieders, D., Wong, K.K.: Fixed Viewpoint Mirror Surface Reconstruction Under an Uncalibrated Camera. IEEE TIP **30**, 2141–2154 (2021)
9. Harris, C.G.: Structure-from-Motion under Orthographic Projection. IJCV **9**(5), 329–332 (1991)
10. Horn, B.K.P., Sjoberg, R.W.: Calculating the Reflectance Map. Applied optics **18**(11), 1770–1779 (1979)
11. Laboratory, T.S.C.G.: The Stanford 3D Scanning Repository, `http://www.graphics.stanford.edu/data/3Dscanrep/`
12. Lagger, P., Salzmann, M., Lepetit, V., Fua, P.: 3D Pose Refinement from Reflections. In: Proc. CVPR (2008)
13. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: LightGlue: Local Feature Matching at Light Speed. In: ICCV (2023)
14. Liu, M., Wong, K.K., Dai, Z., Chen, Z.: Pose estimation from reflections for specular surface recovery. In: Proc. ICCV (2011)
15. Liu, Y., Wang, P., Lin, C., Long, X., Wang, J., Liu, L., Komura, T., Wang, W.: NeRO: Neural Geometry and BRDF Reconstruction of Reflective Objects from Multiview Images. ACM TOG (2023)
16. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. IJCV **60**(2), 91–110 (2004)
17. Matusik, W., Pfister, H., Brand, M., McMillan, L.: A Data-Driven Reflectance Model. ACM TOG **22**(3), 759–769 (2003)

18. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In: Proc. ECCV (2020)
19. Netz, A., Osadchy, M.: Recognition Using Specular Highlights. TPAMI **35**(3), 639–652 (2013)
20. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable Volumetric Rendering: Learning Implicit 3D Representations without 3D Supervision. In: Proc. CVPR (2020)
21. van den Oord, A., Li, Y., Vinyals, O.: Representation Learning with Contrastive Predictive Coding. arXiv preprint arXiv:1807.03748 (2018)
22. Park, K., Patten, T., Vincze, M.: Pix2Pose: Pixel-Wise Coordinate Regression of Objects for 6D Pose Estimation. In: Proc. ICCV (2019)
23. Rematas, K., Ritschel, T., Fritz, M., Gavves, E., Tuytelaars, T.: Deep Reflectance Maps. In: Proc. CVPR (2016)
24. Schnieders, D., Wong, K.K.: Camera and Light Calibration from Reflections on a Sphere. CVIU **117**(10), 1536–1547 (2013)
25. Schönberger, J.L., Frahm, J.M.: Structure-from-Motion Revisited. In: Proc. CVPR (2016)
26. Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson, A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng, Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro, A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods **17**, 261–272 (2020)
27. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. In: Proc. NeurIPS (2021)
28. Xu, Z., Sunkavalli, K., Hadap, S., Ramamoorthi, R.: Deep Image-Based Relighting from Optimal Sparse Samples. ACM TOG **37**(4), 126 (2018)
29. Yamashita, K., Enyo, Y., Nobuhara, S., Nishino, K.: nLMVS-Net: Deep Non-Lambertian Multi-View Stereo. In: Proc. WACV (2023)
30. Yamashita, K., Nobuhara, S., Nishino, K.: DeepShaRM: Multi-View Shape and Reflectance Map Recovery Under Unknown Lighting. In: Proc. 3DV (2024)
31. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview Neural Surface Reconstruction by Disentangling Geometry and Appearance. In: Proc. NeurIPS (2020)
32. Zaal, G., Tuytel, R., Cilliers, R., Ray Cock, J., Mischok, A., Majboroda, S.: Poly Haven, `https://polyhaven.com/hdris`
33. Zhang, K., Luan, F., Wang, Q., Bala, K., Snavely, N.: PhySG: Inverse Rendering with Spherical Gaussians for Physics-based Material Editing and Relighting. In: Proc. CVPR (2021)
34. Zhang, X., Srinivasan, P.P., Deng, B., Debevec, P., Freeman, W.T., Barron, J.T.: NeRFactor: Neural Factorization of Shape and Reflectance Under an Unknown Illumination. ACM TOG **40**(6), 237:1–237:18 (2021)
35. Özyeşil, O., Voroninski, V., Basri, R., Singer, A.: A Survey of Structure from Motion. Acta Numerica **26**, 305–364 (2017)